

1/8

100

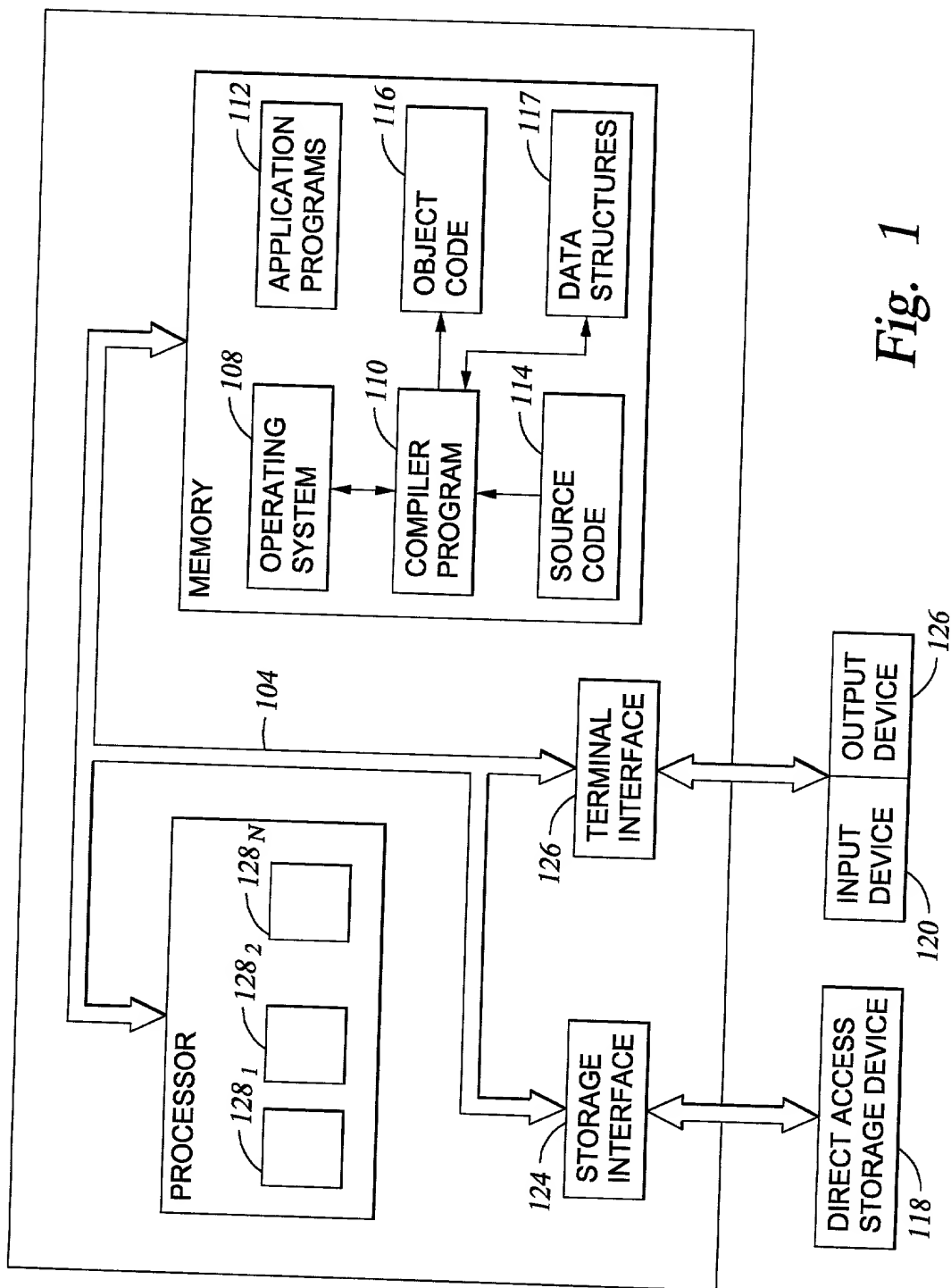


Fig. 1

2/8

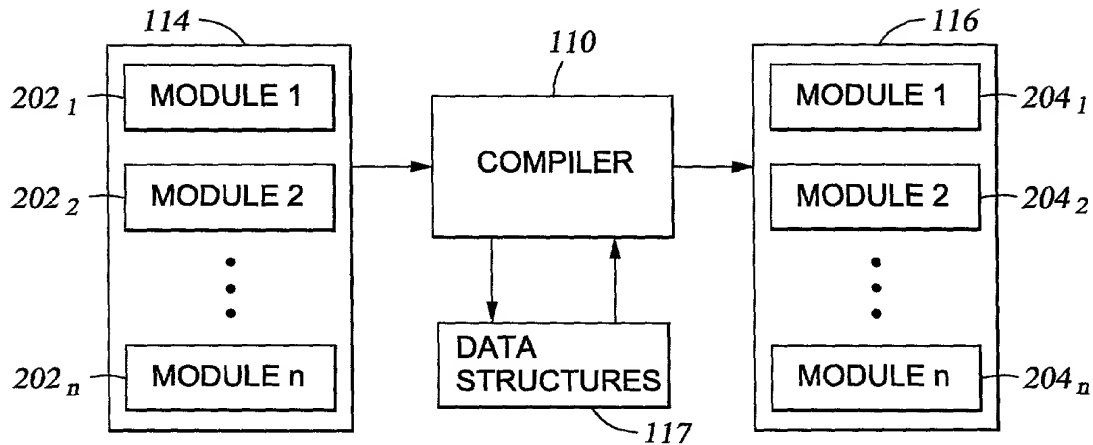


Fig. 2

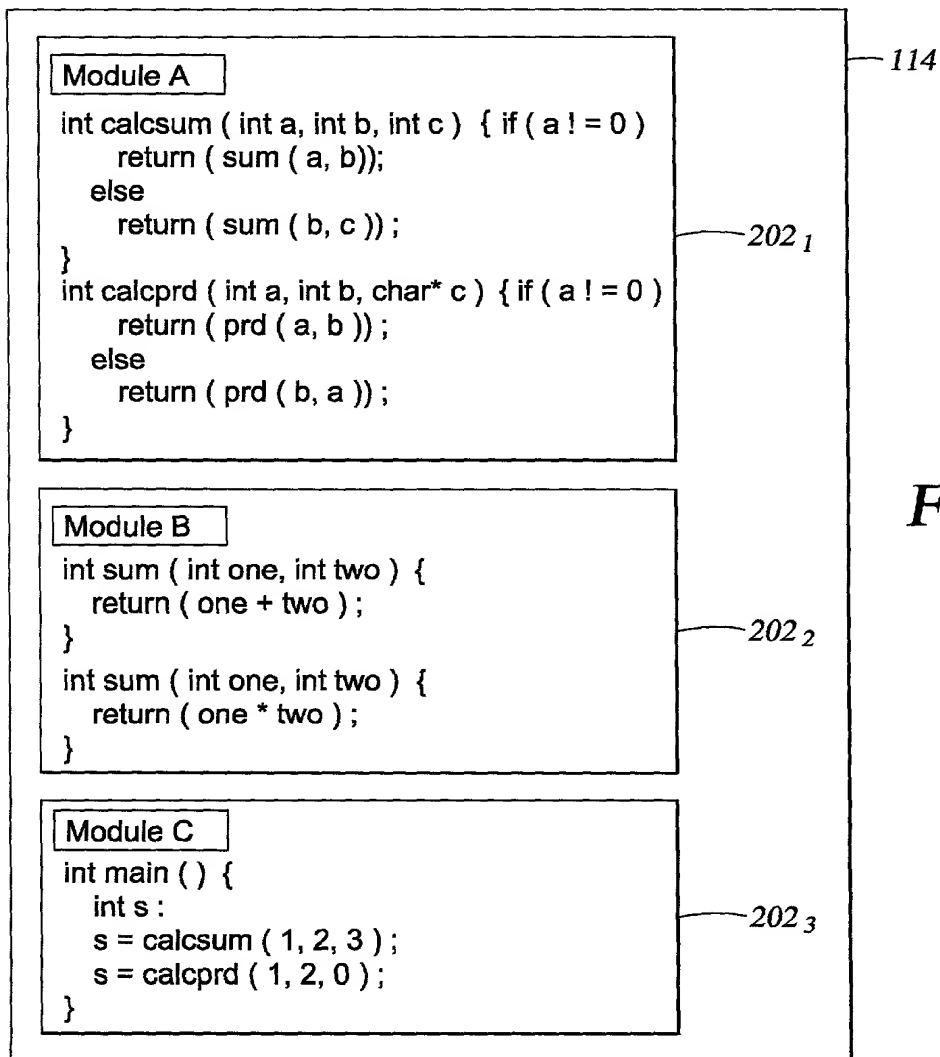


Fig. 3

3/8

402 404 406 408 410 117

Procedure Unique ID #	Module Name	Procedure Name	Argument Count	Argument Types/Lengths
1	A	calsum	3	INTEGER/8, INTEGER/8, INTEGER/8
2	A	calcprd	3	INTEGER/8, INTEGER/8, CHARPTR/8
3	B	sum	2	INTEGER/8, INTEGER/8
4	B	prd	2	INTEGER/8, INTEGER/8
5	C	main	0	INTEGER/8, INTEGER/8

Fig. 4A

412 414 416 418 420 117

Call Link ID #	Caller Unique ID #	Callee Unique ID #	Argument Count	Argument Types/Lengths
1	1	3	2	INTEGER/8, INTEGER/8,
2	1	3	2	INTEGER/8, INTEGER/8,
3	2	4	2	INTEGER/8, INTEGER/8
4	2	4	2	INTEGER/8, INTEGER/8
5	5	1	3	INTEGER/8, INTEGER/8, INTEGER/8
6	5	2	3	INTEGER/8, INTEGER/8, CHARPTR/8

Fig. 4B

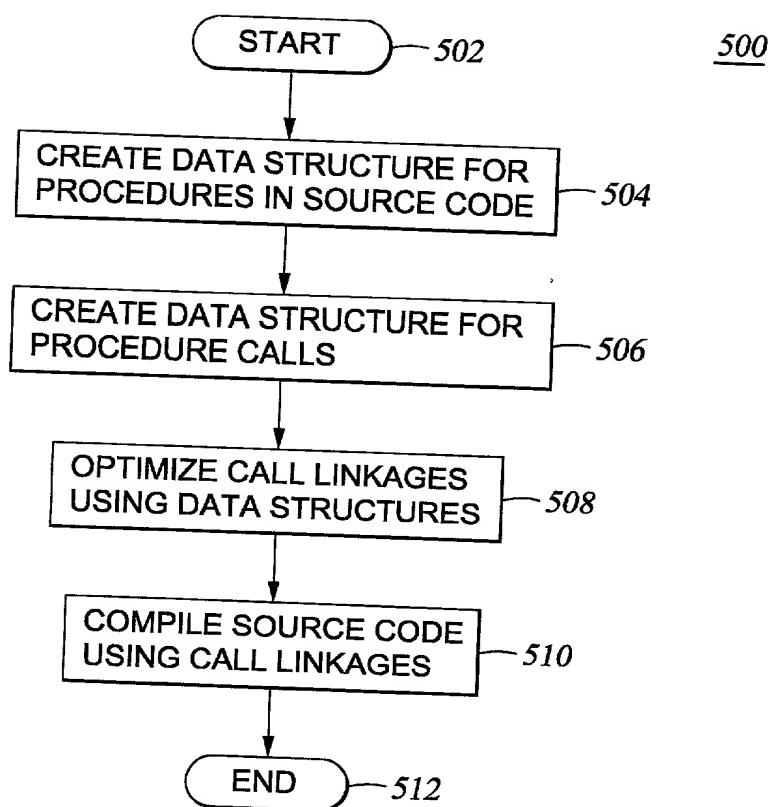


Fig. 5

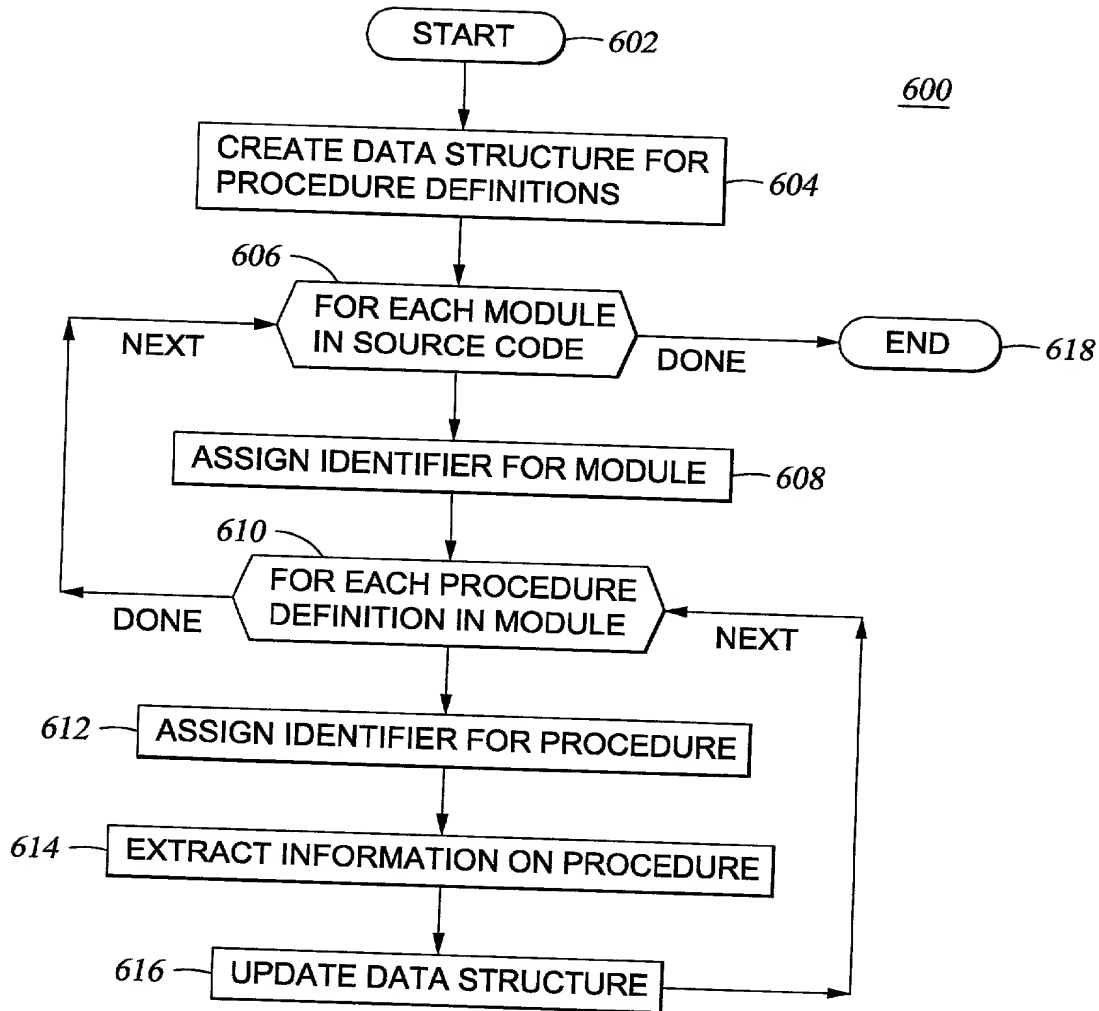


Fig. 6

6/8

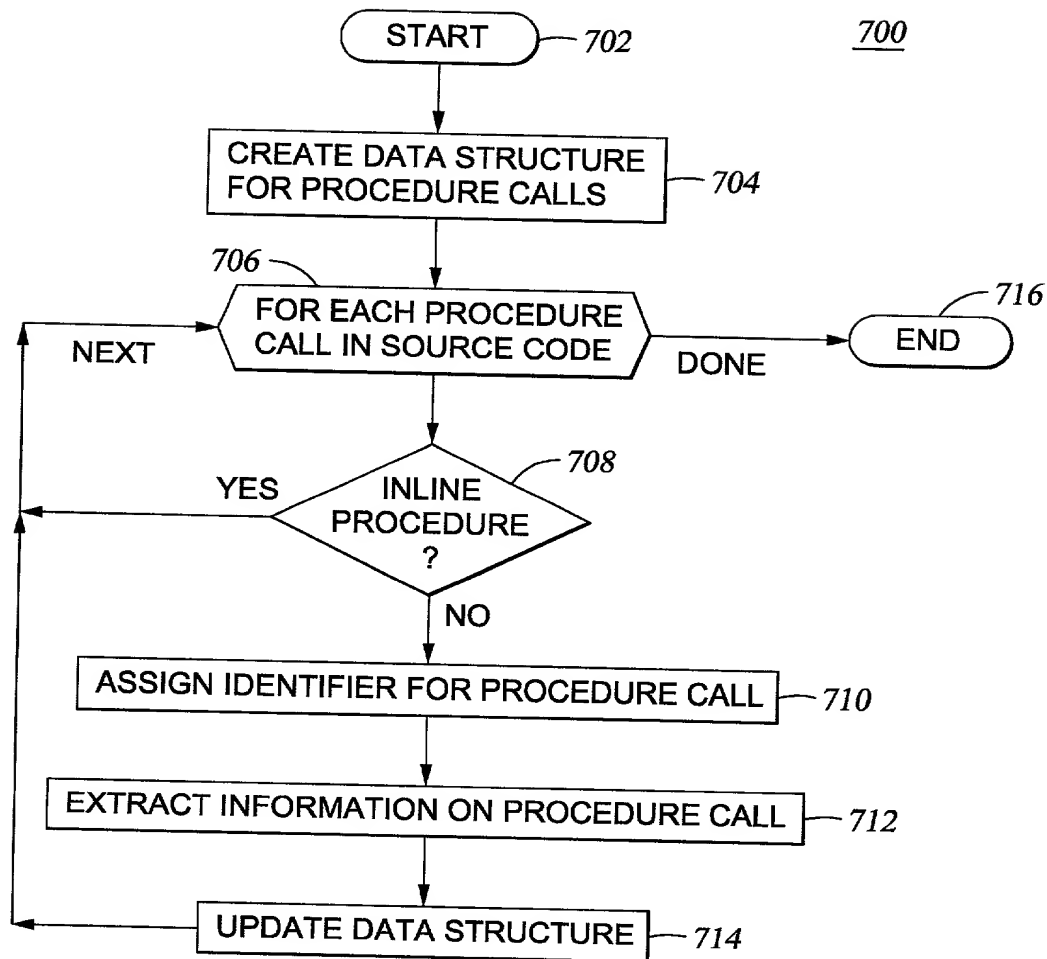


Fig. 7

7/8

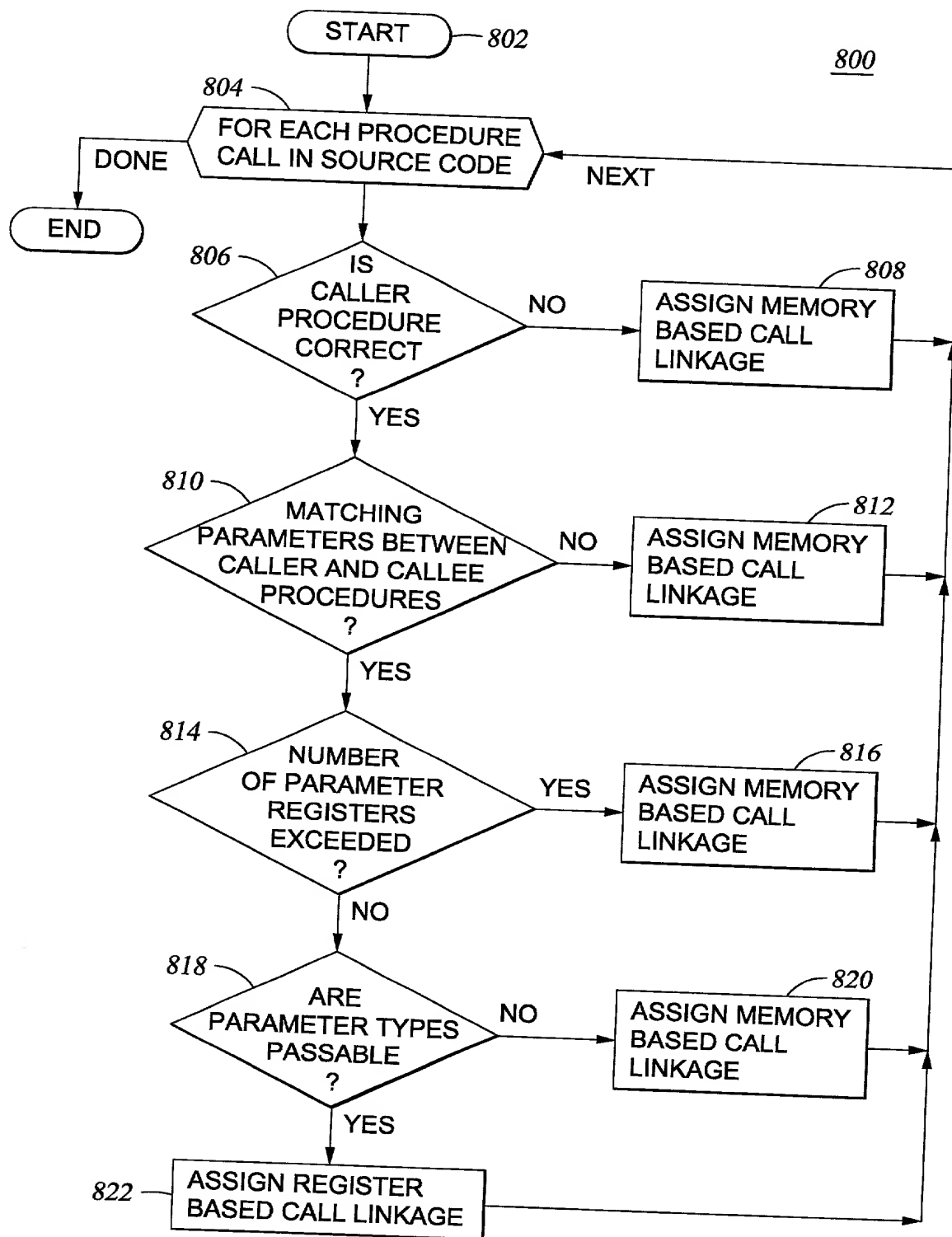


Fig. 8

Procedure Call: prod (a, b)
Callee Procedure: int prod (int one, int two)

Fig. 9A

Allocate memory for parameters (controlblock)
Store those parameters in memory (already within registers)
 STORE a from register to controlblock
 STORE b from register to controlblock +8
Load controlblock register with address of control block
Branch to callee procedure
Retrieve address of control block from controlblock register
 LOAD one to register from controlblock
 LOAD two to register from controlblock +8
Continue on using values within registers

Fig. 9B

Copy parameters into parameter register (already within registers)
 COPY a from register to parameter register 1
 COPY b from register to parameter register 2
Branch to callee procedure
 COPY one to register from parameter register 1
 COPY two to register from parameter register 2
Continue on using values within registers

Fig. 9C